Volume 4, Issue 2, April 2017 ISSN: 2348-8565 (Online)
IJMERT
International Journal of Modern Engineering and Research Technology
Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

VLSI Implementation and Optimization of UDP/IP Stack in FPGA

Abhishek Sahu
M. Tech. Research Scholar
Lakshmi Narain College of Technology
Jabalpur (M.P.), [INDIA]
Email: abhi28686@gmail.com

Prajyant Pathak
Head of the Department
Department of Electronics & Communication Engg.
Lakshmi Narain College of Technology
Jabalpur (M.P.), [INDIA]
Email: prajyant.pathak@yahoo.com

ABSTRACT

This paper is about UDP/IP (user datagram protocol / internet protocol) stack protocol in FPGA (field programmable gate array), This paper presents a proposal of network stack in FPGA, which is the stack of the widely used in VoIP and Video-conference applications. This network node implements the Network, Transport and Link Layer of a traditional stack. This architecture is integrated and developed using Xilinx ISE tool and synthesized to a Spartan-3E FPGA. We show architecture details, timing and area results of a practical prototyping. Also, we compare our prototype and results with other works in terms of area (Xilinx slices), speed (MHz), maximum Ethernet frame length (bytes) and maximum Ethernet speed (Mbps). Comparing to these works our architecture obtained a intermediate solution in area and is the best implementation in terms of speed (MHz).

Keywords:— *FPGA, Network Protocols, OSI Layer, UDP/IP.*

I. INTRODUCTION

Nowadays, the great need of communication in society has collaborated to appear news forms of communication, that are more accessible and lower cost, for example Voice over IP (VoIP) or video conference. But, in a microprocessor of general purpose, this applications compete equally in processing time with other applications, causing a overload in the processing. In order to solve this problem, solutions implemented in dedicated hardware, ASICs or FPGAs become available. This solutions allow that part of the processing, instead of being realized by the microprocessor of general purpose, now can be executed by a dedicated hardware. It belongs to the TCP/IP family, and thus, it is commonly used in conjunction with the IP protocol, replacing the TCP features of the suite. It was obvious for the computer engineers of the time that the network bandwidth was the chokepoint of computing, and the easiest way to increase the bandwidth was to utilize the already available networks in a more optimal way. The TCP protocol allowed stability and had many mechanisms for ensuring data integrity, but the additional checks and communication rules added to the cost of bandwidth. On the other hand, the UDP protocol was designed to be a lot less reliable and relieved from most TCP communication rules which gave the software and hardware engineers more freedom with how they could approach their objectives

II. NETWORK PROTOCOL

Protocol is a set of rules that govern data communications. A protocol defines what is communicated, how is communicated and when it is communicated. The key elements of protocols are syntax, semantics and timing [5]. There are various types of protocol slike TELNET (Telecommunication Network), FTP (file transfer protocol), SMTP (simple mail transfer protocol) and DNS (domain name system) come under application layer of OSI (open system interconnection) model, UDP and TCP come under transport layer and IP and ARP come under network layer.

Usually Internet protocol delivers messages to destination which is selected by unique IP address. ICMP (Internet control message protocol) is used by network devices like routers to send error messages. It is nothing but a network diagnostics protocol and is used to report problems.

III. OSI MODEL

An open system is a set of protocols that allows any two systems to communicate without requiring changes to the logic of the underlying hardware and software. It is not a protocol it is just a theoretical model for understanding and designing a network architecture. It consists of seven layers which are related to each other, which defines a part of process of moving information across a network. Each layer serves the layer above it and served by the layer below it.

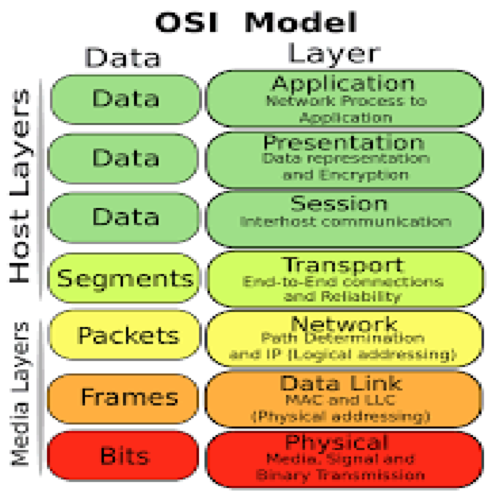

Figure 1. Layers of OSI model

3.1 Physical Layer

It deals with the mechanical and electrical specifications of the data connection. It also defines relationship between device and physical transmission medium. It establishes and terminates the connection between two directly connected nodes, it is flow controlling layer. It predicts transmission mode that is whether simplex, half duplex or full duplex.

3.2 Data link layer

It transforms the physical layer to a reliable link and exchanges data within networks by detecting and possibly correcting errors which may occur in physical layer. Data link packet frame is the basic unit of data transfer for this layer. Frame carries the destination and source link address and other control information in the header.

3.3 Network layer

It is responsible for source to destination delivery of the packet across multiple networks, whereas the data link layer oversees the delivery of packet within same networks. If two systems are connected to the same link, there is no need of network layer. IP is the most important protocol of this layer. It converts logical network address to physical machine address.

3.4 Transport layer

It is responsible for process to process delivery of the entire message. A process is an application program running on the host. It treats each packet independently. The transport layer ensures that the whole message arrives intact and in order, overseeing both error control and flow control from the source to destination. The TCP is the most used protocol of transport level which gives connection oriented communication. Another transport layer protocol is the UDP which provides a unreliable and connectionless communication service.

3.5 Session layer

For some applications service provided by the first three layers (physical, data link and network) is not sufficient then we go foe session layer. It is the network dialog controller, establishes, maintains, and synchronizes the interaction among communicating systems.

3.6 Presentation layer

It deals with the syntax and semantics of the information which is going to exchange between two systems and also responsible for translation, compression, and encryption.

3.7 Application layer

The application layer is closest to end user and enables the user, whether the human or software to access the network. It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared data base management.

4. UDP OVER TCP

TCP provides connection oriented, reliable, full-duplex while UDP provides connectionless, unreliable service. UDP offers minimal datagram delivery service. For sending small messages, UDP it takes much less interactions between sender and receiver than using TCP. It is simpler than TCP.UDP is one of the core member of internet protocol. It provides checksum for data integrity and port number for addressing different functions.

4.1 Uses

- Suitable for the process that requires simple request response communication.
- Used for multicasting and management process.
- Used for some route updating protocols.

V. IMPLEMENTATION

The proper block diagram of UDP/IP stack is as shown in the figure 2.vThe layers, transport, network, data link in UDP/IP stack are designed using VHDL.

5.1 Control transmitter/ receiver

It receives the packet from application and stores it in the RAM transmitter. Control transmitter is for sending data to UDP transmitter block.

In RAM receiver control receiver writes the data in packets from UDP receiver and then sends to the application layer.[2]

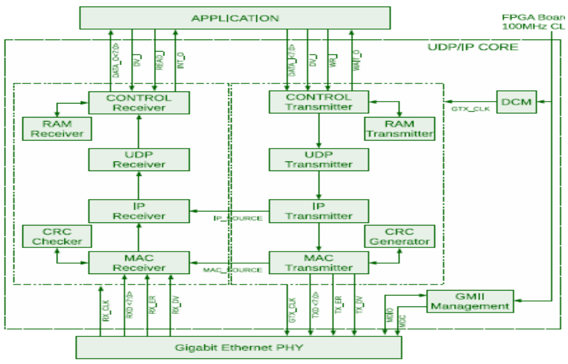

Figure 2-Block Diagram of UDP/IP core

5.2 UDP transmitter/ receiver

It manages UDP packets and represents transport layer. UDP transmitter encapsulates the packet with the UDP header and sends out to the block IP transmitter. UDP receiver checks the packet and sends it to the control receiver without UDP header information. [2]

5.3 IP transmitter / receiver

IP transmitter represents the network layer and manages IPv4 packets. It calculates checksum and encapsulates packet with IP header. IP receiver verifies the checksum of the packet and the destination IP address. Only IP addresses that matches with the core's IP address and broadcast IP address are accepted and send to UDP receiver and others will get discarded. [2]

5.4 MAC transmitter / receiver

MAC transmitter represents the link layer and manages outgoing and incoming of the packets. It sends packet to the physical layer and preamble, where the last nibble is start of the frame delimiter is sent at the beginning. MAC transmitter sets the control signal high. Each byte is sent to the CRC (cyclic redundancy check) generator which progressively calculates the CRC. When packet end is reached the calculated 32-bit CRC is sent.

MAC receiver checks for new packet, when the new packet is detected it will be sent to the CRC checker which will progressively calculates the CRC checksum.

5.5 ARP

ARP(address resolution protocol) block is used to convert network address to physical address and manages ARP packets. It allows the MAC address from other node when only IP address of its neighbors is known. It uses simple message format that contains one address resolution request.

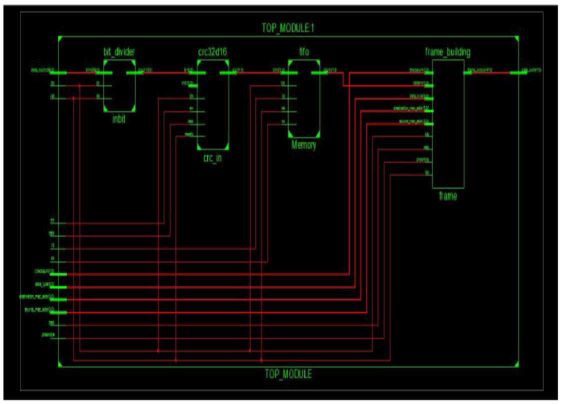

Figure 3:Top Module RTL Schematic Diagram

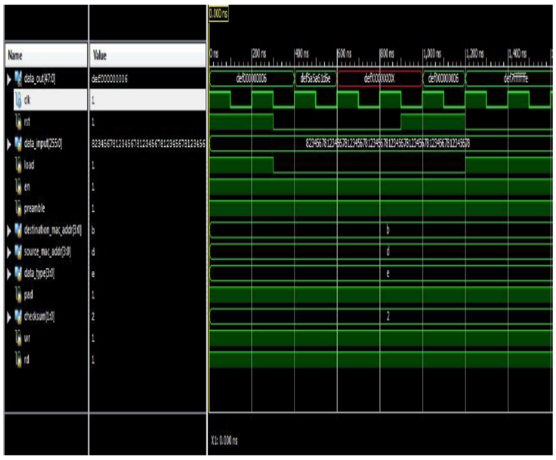

Figure 4:Transmitter module simulation results

VI. Comparative Result

Table 1 : Comparative Result

Parameter	**Nikolaos Alachiotis, Simon A. Berger[1]**	**Result Obtained**
Slice Register	184	82
FMax(Mhz)	128.8	161.91

VII. Future Scope

We propose to improve UDP protocol so that the physical frame error indication is forwarded to the application for better error control. Various optimization techniques can be implemented to reduce area, increase speed. Timing constrains can be and data integrity. We can also implement different types of UDP like CUDP, UDP-Lite for real time multimedia applications over wireless networks.

REFERENCES:

[1] A. Lofgren, L. Lodesten, S. Sjoholm, and H. Hansson "An Analysis of FPGA-based UDP/IP Stack Parallelism for Embedded Ethernet Connectivity".

[2] K. Morita and K. Abe. "Implementation of UDP/IP Protocol Stack on FPGA and its Performance Evaluation".

[3] Fernandoluis Herrmann, Guilherme Perin, Josue Paulo Jose de Freitas, Rafael Bertagnolli and Joao Baptista dos Santos Martins "An UDP/IP Network Stack in FPGA".

[4] Weidong Lu "Designing TCP/IP Functions in FPGAs", MSc thesis, code number C-MS-2003-09.

[5] Behrouz A. Forouzandata "Communications and Networking IV Edition Mc-Graw Hill Publications.

[6] Andrew S. Tanenbaum "Data Communication and Networking".

[7] Aditya Prakash Chaubal "Design and Implementation of an FPGA-based Partially Reconfigurable Network Controller.

[8] Haitao Zheng, Jill Boyce "An improved UDP protocol for video transmission over internet-to-wireless networks".

[9] Nikolaos Alachiotrs, Simon A. Berger, Alexandros Stamatakis A versatile UDP/IP based pc <-> FPGA communication platform.

[10] Puneet Kaur "Performance Comparison of Transport Protocols, UDP and UDP-Lite for Transmission of Different Video Codes Over MANETs."

* * * * *